

Forum:[Hack Forums](#)

Thread: [Grab region of screen](#)

quartzd - February 20, 2016, 6:58 am

Hi im not really sure how to explain this problem but i want to be able to grab pixels from a rectangle region on a screen at position x1,y1 and x2,y2 and copy those exact pixels onto a form... hard to explain I think

Here is a basic image example

It would look like that..where the red rectanagle region is copied into the form as the image shows..
does anyone know how I could do this?

i.imgur.com/iP8ilwp.jpg

(can't post links....)

If anyone can post the image here for me that would be nice thanks

SubGenius - February 20, 2016, 7:37 am

The image OP posted:

Spoiler [\(Click to View\)](#)

So, where are you stuck? You can't replicate that selection window or don't know how to crop an image?

If the second, the easiest way to do it is capture a full screen screenshot, then based on the location and size of the selection form (red rectangle) you can crop a new image from the full screenshot.

quartzd - February 20, 2016, 7:50 am

The image wasn't real..it was just to show what the end result should like like but where I am stuck is how to replicate a region of a window...I don't know how to go about doing it

I don't know which API to use or once I do grab a region of a window..how to post that exact region onto a form or panel

Or I don't know the most efficient way of doing it...I don't know exactly what that technique is called

AnonCensored - February 20, 2016, 8:48 am

(Feb 20, 2016 08:50 AM)quartzd Wrote: [\[CITE\]](#) The image wasn't real..it was just to show what the end result should like like but where I am stuck is how to replicate a region of a window...I don't know how to go about doing it

I don't know which API to use or once I do grab a region of a window..how to post that exact region onto a form or panel

Or I don't know the most efficient way of doing it...I don't know exactly what that technique is called

First off, You don't need to use a api at all to do this, second off it's quite easy

There's many ways you can go about this by using a picturebox and load the image or load the image on the form etc..

First you'd capture a full desktop image then display it in the form or picturebox

PHP Code

```
<?php  
this.BackgroundImage == Imagehere
```

Then you'll want to take change the borders to none and and adjust the size to your screen size

PHP Code

```
<?php  
this.FormBorderStyle = Windows.Forms.FormBorderStyle.None;  
this.Size = Screen.PrimaryScreen.Bounds.Size;
```

Then you'd start to draw

example

PHP Code

```
<?php  
using Microsoft.VisualBasic;  
using System;  
using System.Collections;  
using System.Collections.Generic;  
using System.Data;  
using System.Diagnostics;  
using System.Math;  
  
public class Form1  
{  
  
    private Rectangle MyRectangle;  
    private bool Drawing = false;  
    private float StartX;  
    private float StartY;  
    private float CursorX;  
  
    private float CursorY;  
    private void YourForm_KeyDown(object sender, System.Windows.Forms.KeyEventArgs e)  
    {  
        if (e.KeyCode == Keys.Escape) {  
            this.Close();  
        }  
    }  
  
    private void YourForm_MouseDown(object sender, System.Windows.Forms.MouseEventArgs e)  
    {  
        Drawing = true;
```

```

    StartX = e.X;
    StartY = e.Y;
    CursorX = e.X;
    CursorY = e.Y;
}

private void YourForm_MouseMove(object sender, System.Windows.Forms.MouseEventArgs e)
{
    if (!Drawing)
        return;
    Graphics G = this.CreateGraphics;
    CursorX = e.X;
    CursorY = e.Y;
    this.Invalidate();
}

private void YourForm_MouseUp(object sender, System.Windows.Forms.MouseEventArgs e)
{
    Drawing = false;
    Graphics G = this.CreateGraphics;
    G.DrawRectangle(Pens.Black, MyRectangle.X, MyRectangle.Y, MyRectangle.Width,
MyRectangle.Height);
    this.Invalidate();
}

private void YourForm_Paint(object sender, System.Windows.Forms.PaintEventArgs e)
{
    if (Drawing == true) {
        Pen Dashed_Pen = new Pen(Brushes.Red, 2);

        Dashed_Pen.DashStyle = Drawing2D.DashStyle.Solid;
        MyRectangle = new Rectangle(Min(StartX, CursorX), Min(StartY, CursorY), Abs(StartX -
CursorX), Abs(StartY - CursorY));

        e.Graphics.FillRectangle(new SolidBrush(Color.Fuchsia), MyRectangle);
        e.Graphics.DrawRectangle(Dashed_Pen, MyRectangle);
    }
}

private void YourForm_Load(System.Object sender, System.EventArgs e)
{
    ScreenShot.ScreenCapture sc = new ScreenShot.ScreenCapture();
    Bitmap b = sc.CaptureScreen;
    this.BackgroundImage = b;

    this.Focus();
}

```

```

this.FormBorderStyle = Windows.Forms.FormBorderStyle.None;
this.TopMost = true;
this.Location = new Point(0, 0);
this.Size = Screen.PrimaryScreen.Bounds.Size;
this.DoubleBuffered = true;

}
public Form1()
{
    Load += Region_Drag_Load;
    Paint += Region_Drag_Paint;
    MouseUp += Region_Drag_MouseUp;
    MouseMove += Region_Drag_MouseMove;
    MouseDown += Region_Drag_MouseDown;
    KeyDown += Region_Drag_KeyDown;
}
}

```

Then from there you could save the cords of the region once mouse click has been pressed then like crop the specific part then save it, You can do the rest i gave you a somewhat idea on how to..

SubGenius - February 20, 2016, 8:51 am

I just spent my last 30 minutes coding this simple app.

get the source here:<http://www.filedropper.com/omgdude>

Although it's written in VB and has no comments at all, it's very simple and you can easily copy and paste the stuff u need.

edit: this is almost exactly what anon above me provided you.

GuRuHaX - February 20, 2016, 9:00 am

(Feb 20, 2016 08:50 AM)quartzd Wrote: [\[CITE\]](#) The image wasn't real..it was just to show what the end result should like like but where I am stuck is how to replicate a region of a window...I don't know how to go about doing it

I don't know which API to use or once I do grab a region of a window..how to post that exact region onto a form or panel

Or I don't know the most efficient way of doing it...I don't know exactly what that technique is called This will get you a complete screenshot of current screen:

PHP Code

```

<?php
using (Bitmap ScreenShot = new Bitmap(Screen.PrimaryScreen.Bounds.Width,
                                        Screen.PrimaryScreen.Bounds.Height))
{
    using (Graphics g = Graphics.FromImage(ScreenShot))
    {
        g.CopyFromScreen(Screen.PrimaryScreen.Bounds.X,
                        Screen.PrimaryScreen.Bounds.Y,
                        0, 0,
                        ScreenShot.Size,
                        CopyPixelOperation.SourceCopy);
    }
};

```

```
        ScreenShot.Save(pathToSave);  
    };
```

I think if you dont fill in path it saves in your Bin/Debug folder as default.

This is for a specific area :

PHP Code

```
<?php  
public void DrawRectangle()  
{  
    Rectangle rect = new Rectangle(0, 0, 100, 100);  
    using (Bitmap bmp = new Bitmap(rect.Width, rect.Height, PixelFormat.Format32bppArgb))  
    {  
        using (Graphics g = Graphics.FromImage(bmp))  
        {  
            g.CopyFromScreen(rect.Left, rect.Top, 0, 0, bmp.Size, CopyPixelOperation.  
SourceCopy);  
            string fileName = "your_Screenshot_Name";  
            bmp.Save(fileName, ImageFormat.Jpeg);  
        }  
    }  
}
```

Where in first line the 100 ,100 values are your Width & Height.

Hope this helps !
?

(Feb 20, 2016 09:48 AM)AnonCensored Wrote: [\[CITE\]](#) First off, You don't need to use a api at all to do this, second off it's quite easy

There's many ways you can go about this by using a picturebox and load the image or load the image on the form etc..

First you'd capture a full desktop image then display it in the form or picturebox

PHP Code

```
<?php  
this.BackgroundImage == Imagehere
```

Then you'll want to take change the borders to none and and adjust the size to your screen size

PHP Code

```
<?php  
this.FormBorderStyle = Windows.Forms.FormBorderStyle.None;  
this.Size = Screen.PrimaryScreen.Bounds.Size;
```

Then you'd start to draw

example

PHP Code

```
<?php
using Microsoft.VisualBasic;
using System;
using System.Collections;
using System.Collections.Generic;
using System.Data;
using System.Diagnostics;
using System.Math;

public class Form1
{

    private Rectangle MyRectangle;
    private bool Drawing = false;
    private float StartX;
    private float StartY;
    private float CursorX;

    private float CursorY;
    private void YourForm_KeyDown(object sender, System.Windows.Forms.KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Escape) {
            this.Close();
        }
    }

    private void YourForm_MouseDown(object sender, System.Windows.Forms.MouseEventArgs e)
    {
        Drawing = true;
        StartX = e.X;
        StartY = e.Y;
        CursorX = e.X;
        CursorY = e.Y;
    }

    private void YourForm_MouseMove(object sender, System.Windows.Forms.MouseEventArgs e)
    {
        if (!Drawing)
            return;
        Graphics G = this.CreateGraphics;
        CursorX = e.X;
        CursorY = e.Y;
        this.Invalidate();
    }

    private void YourForm_MouseUp(object sender, System.Windows.Forms.MouseEventArgs e)
    {
        Drawing = false;
        Graphics G = this.CreateGraphics;
        G.DrawRectangle(Pens.Black, MyRectangle.X, MyRectangle.Y, MyRectangle.Width,
```

```

MyRectangle.Height);
    this.Invalidate();

}

private void YourForm_Paint(object sender, System.Windows.Forms.PaintEventArgs e)
{
    if (Drawing == true) {
        Pen Dashed_Pen = new Pen(Brushes.Red, 2);

        Dashed_Pen.DashStyle = Drawing2D.DashStyle.Solid;
        MyRectangle = new Rectangle(Min(StartX, CursorX), Min(StartY, CursorY), Abs(StartX -
CursorX), Abs(StartY - CursorY));

        e.Graphics.FillRectangle(new SolidBrush(Color.Fuchsia), MyRectangle);
        e.Graphics.DrawRectangle(Dashed_Pen, MyRectangle);
    }
}

private void YourForm_Load(System.Object sender, System.EventArgs e)
{
    ScreenShot.ScreenCapture sc = new ScreenShot.ScreenCapture();
    Bitmap b = sc.CaptureScreen;
    this.BackgroundImage = b;

    this.Focus();
    this.FormBorderStyle = Windows.Forms.FormBorderStyle.None;
    this.TopMost = true;
    this.Location = new Point(0, 0);
    this.Size = Screen.PrimaryScreen.Bounds.Size;
    this.DoubleBuffered = true;

}
public Form1()
{
    Load += Region_Drag_Load;
    Paint += Region_Drag_Paint;
    MouseUp += Region_Drag_MouseUp;
    MouseMove += Region_Drag_MouseMove;
    MouseDown += Region_Drag_MouseDown;
    KeyDown += Region_Drag_KeyDown;
}
}

```

Then from there you could save the cords of the region once mouse click has been pressed then like crop the specific part then save it, You can do the rest i gave you a somewhat idea on how to..

(Feb 20, 2016 09:51 AM)SubGenius Wrote: [\[CITE\]](#) I just spent my last 30 minutes coding this simple app.

get the source here:<http://www.filedropper.com/omgdude>

Although it's written in VB and has no comments at all, it's very simple and you can easily copy and paste the stuff u need.

edit: this is almost exactly what anon above me provided you.

OR WHAT THEY SAY ?

Didn't now i was writing so slow there was only one reply when i started ?

?

quartzd - February 20, 2016, 2:38 pm

Thanks all for the replies, that is what i was looking for